IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

An Intelligent IoT Framework for Battery Health Monitoring and Remaining Useful Life Prediction

1Kamaljeet Kour, 2Er. Sheetal Gandotra, 3Dr. Mekhla Sharma

1M.Tech Scholar, 2Associate Professor , 3Assistant Professor

1Government College of Engineering and Technology (GCET), Jammu, India,

2Government College of Engineering and Technology (GCET), Jammu, India,

3Government College of Engineering and Technology (GCET), Jammu, India

Abstract: Lithium-ion batteries are the preferred energy storage technology across electric vehicles, renewable energy systems, and portable electronics, largely because of their high energy density, long cycle life, and dependable performance. Repeated charging and discharging, together with temperature swings and varying operating conditions, gradually wear the cells down, so knowing a battery's State of Health (SOH) and Remaining Useful Life (RUL) at any given moment matters a great deal for safety, reliability, and efficient use. Conventional model-based and machine learning techniques tend to lose accuracy once battery degradation departs from the neat nonlinear assumptions built into them, particularly under changing operating conditions. This paper proposes a deep learning framework that predicts SOH and RUL together, using an Attention-Based Bidirectional Long Short-Term Memory (Attention-BiLSTM) network paired with an Internet of Things (IoT)-enabled Battery Management System (BMS). Historical degradation data is combined with real-time voltage, current, temperature, and capacity readings collected through IoT sensors, cleaned, normalized, and reshaped into sequences before training. The bidirectional structure lets the model draw on both past and future context within a sequence, while the attention layer highlights the time steps that matter most for degradation. Benchmarked against CNN-LSTM, BiLSTM, GRU, and conventional LSTM baselines using MAE, RMSE, and R^2, the Attention-BiLSTM model came out ahead on every metric, achieving an MAE of 0.5489, an RMSE of 0.6861, and an R^2 of 0.8436. The accompanying IoT-enabled Smart BMS prototype, built around an Arduino UNO with voltage, current, and temperature sensing, further demonstrated that continuous monitoring, fault detection, and predictive maintenance can be delivered within a single practical system suitable for electric vehicles, smart grids, and renewable-energy storage.

Keywords – Lithium-Ion Battery, State of Health (SOH), Remaining Useful Life (RUL), Attention-BiLSTM, CNN-LSTM, GRU, Deep Learning, Internet of Things (IoT), Battery Management System (BMS), Predictive Maintenance.

I. INTRODUCTION

Lithium-ion batteries have become the default energy storage technology across modern electrical and electronic systems, and it is not hard to see why: high energy density, a light footprint, low self-discharge, and a long operating life make them well suited to electric vehicles (EVs), hybrid electric vehicles (HEVs), renewable energy storage, aerospace systems, medical devices, consumer electronics, and industrial automation alike. As efforts to cut greenhouse gas emissions and expand clean energy adoption continue to gather pace, the demand placed on these battery systems to be efficient, reliable, and above all safe keeps growing. That reliability, though, is not something batteries hold onto indefinitely. Repeated charge discharge cycling, high operating temperatures, overcurrent conditions, and general environmental wear all chip away at a cell's ability to store and deliver energy. In practical terms, this shows up as reduced driving range in EVs, shorter runtime in portable devices, rising maintenance costs, and, in the worst cases, an elevated risk of sudden failure. Because of this, continuous health monitoring has become one of the core responsibilities of any modern Battery Management System (BMS). Two quantities are typically used to describe battery condition in a way that is actually useful for decision-making. State of Health (SOH) captures how a battery's current condition compares with its original, fresh state, generally expressed as the ratio of present maximum capacity to rated capacity- an SOH close to 100% means a healthy battery, while lower values point to progressive ageing. Remaining Useful Life (RUL), meanwhile, estimates how many charge–discharge cycles or how much operating time remain before the battery crosses its end-of-life threshold, conventionally set at 80% of initial capacity. Reliable SOH and RUL estimates are what make predictive maintenance, failure prevention, and extended service life possible in the first place. Traditional battery health estimation has relied on electrochemical models, equivalent circuit models, physics-based approaches, and filtering techniques such as Kalman or particle filters. These methods can offer real insight into battery behaviour, but they typically demand detailed domain knowledge, precise parameter identification, and non-trivial computation and their accuracy tends to slip once operating conditions move away from the assumptions baked into the model, which is a real problem given how nonlinear and multi-factor battery degradation actually is.

Deep learning offers a different way in: rather than encoding a physical model by hand, a network can learn the relevant degradation relationships directly from historical data [22]. Long Short-Term Memory (LSTM) networks, in particular, have proven effective for this kind of time-series problem because their gating mechanism lets them retain long-term dependencies while avoiding the vanishing-gradient issues that limit conventional recurrent networks [20]. Bidirectional LSTM (BiLSTM) pushes this further by processing a sequence in both forward and backward directions, giving the model a fuller picture of the degradation pattern than a one-directional LSTM can offer on its own, and simplified recurrent variants such as the Gated Recurrent Unit (GRU) and hybrid CNN-LSTM architectures have both been explored as ways of trading off accuracy against computational cost. Even with bidirectional processing, though, not every point in a degradation sequence carries equal weight. This is the gap an attention mechanism is designed to close it lets the network assign higher importance to the time steps that actually matter for the prediction, rather than treating the whole sequence uniformly. Combining attention with BiLSTM, as proposed here, allows the model to concentrate on the most informative degradation features, which in practice tends to mean better accuracy, faster convergence, and steadier performance across different operating conditions.

Alongside these modelling advances, the Internet of Things (IoT) has changed what is practically possible in battery monitoring. An IoT-enabled BMS can continuously acquire voltage, current, temperature, and other parameters through embedded sensors and microcontrollers, and stream that data to a local or cloud platform for visualization and analysis. This opens the door to early detection of abnormal conditions, remote diagnostics, and timely safety interventions and when paired with a deep learning model, it becomes possible to move from passive monitoring to genuinely predictive, decision-supporting battery management.This paper brings these pieces together: an IoT-enabled BMS is integrated with an Attention-BiLSTM deep learning model to jointly predict SOH and RUL for lithium-ion batteries, and its performance is benchmarked directly against CNN-LSTM, BiLSTM, and GRU baselines trained under identical conditions. Battery measurements drawn from a publicly available ageing dataset, together with real-time sensor readings, are cleaned, normalized, feature-extracted, and converted into sequences before being used to train the prediction models.

The main contributions of this work can be summarised as follows: (1) an IoT-enabled BMS is developed for continuous acquisition of voltage, current, temperature, and capacity, complete with a working hardware prototype; (2) an Attention-BiLSTM model is proposed for joint SOH and RUL estimation, learning temporal degradation patterns directly from data; (3) a complete preprocessing pipeline- cleaning, normalization, feature extraction, and sequence generation- supports stable model training; (4) the proposed model is benchmarked against CNN-LSTM, BiLSTM, GRU, and conventional LSTM using MAE, RMSE, and R^2; and (5) a working Smart BMS hardware prototype demonstrates that the approach is practically deployable, not just theoretically accurate.

II. LITERATURE REVIEW

Accurate SOH and RUL prediction has drawn sustained research interest as lithium-ion batteries have moved from consumer gadgets into electric vehicles and grid-scale storage, where the cost of getting a prediction wrong is much higher. Early work in the field leaned on physics-based and equivalent-circuit models: electrochemical models describe the internal chemistry and degradation mechanisms in detail [3], [4], but they require extensive knowledge of battery chemistry and heavy computation, which limits their use in real-time BMS applications, while equivalent circuit models are computationally lighter but often fail to represent complex ageing behaviour under varying conditions. Data-driven approaches followed, treating degradation as something that could be learned from historical operating data rather than derived from first principles. Techniques such as Support Vector Regression, Artificial Neural Networks, Random Forest, and Gaussian Process Regression [5], [6] improved on purely mathematical models but still generally depended on manually engineered features and struggled to capture long-range sequential dependencies in the degradation data. Even relatively simple data-driven models have been shown to predict battery cycle life well before capacity fade becomes clearly visible, provided the right early-cycle features are chosen [25], which underlines how much predictive signal is already present in the kind of early operating data used in this study.

The arrival of deep learning shifted this further. Convolutional Neural Networks were applied to automatically extract useful features from battery measurements, though their strength lies in spatial rather than temporal structure, which limits how well they capture long-term dependencies on their own; hybrid CNN-LSTM architectures were subsequently proposed to combine this feature-extraction strength with genuine temporal learning [13]. Recurrent architectures more broadly have proved a good fit for this kind of sequential problem: Long Short-Term Memory networks address the vanishing-gradient limitations of conventional RNNs through their gating mechanism and have been shown to estimate SOH and RUL with clear accuracy gains over traditional machine learning [9], and related recurrent formulations have been used for joint state-of-charge and state-of-health estimation as well [15]. Because a standard LSTM only processes information in one direction, however, it can miss degradation information available from the opposite temporal direction a gap that Bidirectional LSTM (BiLSTM) networks were introduced to close, processing sequences forward and backward simultaneously and, according to several studies, improving RUL prediction performance as a result [16], [21].

Even with bidirectional processing, not every point in a degradation sequence is equally informative, which is the motivation behind attention-based architectures [19]. By assigning different importance weights across time steps, attention layers let a model focus on the degradation events that matter most while downplaying less relevant fluctuations, and hybrid Attention-BiLSTM and Attention-LSTM architectures have reported improved accuracy and generalisation on comparable time-series prediction problems [17], [18]. A broader body of related work has also explored neural-network-based and uncertainty-aware formulations of the RUL prediction problem more generally [10]–[12], [14], and recent survey work continues to track how these deep learning approaches compare against one another [8], [24]. In parallel, IoT technologies have reshaped what is practically achievable in battery monitoring, with sensor- and microcontroller-based platforms enabling continuous acquisition of voltage, current, and temperature and supporting remote diagnostics and predictive maintenance [7].

What is comparatively rare in the literature, however, is a framework that brings both strands together advanced attention-based temporal learning and real-time IoT-based acquisition into a single working system, rather than treating offline prediction and real-time monitoring as separate problems. Table I summarises this progression across representative categories of prior work.

Table I. Summary of Existing Research on Lithium-Ion Battery Health Prediction

Approach	Representative Reference	Dataset Type	Major Contribution	Limitation
LSTM Network	[9]	Public battery ageing dataset	Demonstrated deep learning capability for SOH/RUL prediction	Learns in one temporal direction only
CNN-LSTM Model	[13]	Public cycling dataset	Combined spatial feature extraction with temporal learning	Higher computational complexity
BiLSTM Model	[16]	NASA-type dataset	Improved sequence-dependency learning via bidirectional processing	No automatic feature weighting
Attention-LSTM	[17]	Battery ageing dataset	Improved feature importance selection via attention	Limited bidirectional context
Classical ML methods	[4], [5]	Experimental battery data	Compared regression-based estimation approaches	Requires manual feature extraction
IoT-Based BMS	[7]	Real-time sensor data	Enabled remote, continuous battery monitoring	Limited predictive intelligence

Research Gap

Taken together, the literature points to a fairly consistent set of gaps. Many studies rely solely on offline datasets without addressing real-time monitoring needs; conventional machine learning still leans on manual feature extraction and struggles with nonlinear degradation; LSTM-based methods capture temporal structure but only in one direction; BiLSTM improves on this but does not automatically identify which degradation features matter most; attention-based models address feature importance but are rarely combined with robust bidirectional temporal learning; and existing IoT-based BMS platforms tend to focus on data acquisition and monitoring rather than intelligent prediction. This study proposes an integrated Attention-BiLSTM framework, benchmarked directly against CNN-LSTM, BiLSTM, and GRU baselines and combined with an IoT-enabled BMS, specifically to close this combination gap-pairing real-time monitoring capability with the temporal and attention-based learning needed for reliable SOH and RUL estimation under practical operating conditions.

III. PROPOSED METHODOLOGY

The proposed framework brings together real-time battery monitoring, structured data preprocessing, temporal feature learning, and intelligent prediction to address the limitations identified above. It is organised around four stages: (1) IoT-based battery data acquisition, (2) data preprocessing and feature engineering, (3) deep learning-based SOH and RUL prediction, and (4) performance evaluation and real-time monitoring. In operation, the system continuously collects battery parameters, processes them, predicts the relevant health indicators, and feeds that information back as decision support for maintenance and safety management.

A. Overall System Architecture

The architecture is organised into four layers. The sensing layer collects voltage, current, temperature, and capacity through sensors connected to an embedded controller, which forms the acquisition core of the IoT-enabled BMS. The data processing layer handles cleaning, missing-value treatment, normalization, feature extraction, and sequence generation, reflecting the fact that battery degradation unfolds over time and needs to be presented to the model as ordered sequences rather than isolated readings. The prediction layer houses the deep learning models themselves, which learn the nonlinear relationship between operating parameters and degradation to estimate SOH and RUL. Finally, the monitoring layer visualises battery condition and predicted health status, letting a user track performance, spot abnormal conditions, and act on them before they become failures. Fig. 1 illustrates this overall system architecture.

B. IoT-Enabled Battery Management System Hardware Design

Beyond the modelling side, a working IoT-enabled BMS prototype was built to track battery condition continuously and act on it in real time. It brings together sensing, processing, display, and protection functions in a single unit: dedicated sensors measure battery voltage, current, and temperature and pass these readings to an Arduino UNO microcontroller, which derives quantities such as state of charge and overall battery status from them. A 0.96-inch OLED screen gives a live, at-a-glance view of these readings, while a relay module, cooling fan, and buzzer provide automatic protection, disconnecting the load, dissipating heat, or sounding an alarm whenever voltage, temperature, or other readings drift outside safe limits. The same sensor data is streamed to a computer over serial communication, where it feeds the deep learning models described later in this section.

1) Arduino UNO

At the centre of the prototype sits an Arduino UNO, based on the ATmega328P microcontroller, which handles real-time sensor acquisition, runs the monitoring logic, and drives the protection devices. Voltage, current, and temperature readings arrive through its analog input pins, are processed into meaningful battery parameters, and are pushed out to the OLED display for the user to see. Table II summarises its key specifications.

Table II. Arduino UNO Specifications

Parameter	Value
Microcontroller	ATmega328P
Operating Voltage	5 V
Clock Frequency	16 MHz
Flash Memory	32 KB
SRAM	2 KB
Analog Inputs	6
Digital I/O Pins	14

2) Voltage Sensing Module

Battery voltage is one of the more telling indicators of operating condition, so a 0–25 V DC voltage sensor module is used to track terminal voltage continuously. Because the Arduino's analog inputs only accept up to 5 V, the sensor uses an internal resistive divider network to bring higher battery voltages within that range. The resulting analog signal is converted by the Arduino into an actual voltage value, which is displayed on the OLED, logged over serial communication, and passed to the prediction model as one of its core input features.

3) Current Sensing Module (ACS712)

Charge and discharge current says a great deal about how a battery is being used, and here it is measured with an ACS712 Hall-effect current sensor, chosen partly because it electrically isolates the sensing circuit from the microcontroller for safer operation. As current flows through the battery circuit, the ACS712's analog output is converted by the Arduino into an actual current value, displayed locally, logged, and fed into the prediction model alongside voltage and, just as importantly, used to trigger the protection logic when current behaviour looks abnormal.

4) Temperature Sensing Module

Because thermal conditions have such a strong bearing on degradation, a temperature sensor connected to analog pin A2 continuously tracks battery temperature. Readings are shown on the OLED, and once temperature crosses a predefined threshold, the Arduino activates the cooling fan and buzzer to protect the cell from thermal damage.

5) OLED Display

A 0.96-inch I2C OLED display gives a real-time, local view of voltage, current, temperature, state of charge, and overall system status, communicating with the Arduino over the I2C bus (SDA to A4, SCL to A5) to keep the wiring simple.

6) Relay Module

A 5 V relay module, switched through digital pin D10, disconnects the battery from its load whenever the system detects an unsafe condition such as over-voltage or excessive temperature, giving the prototype a hard cut-off in addition to its predictive monitoring.

7) Cooling Fan

A small 5 V DC cooling fan, driven through digital pin D9, is activated automatically once measured temperature exceeds the safety threshold, helping bring cell temperature back down before it becomes a hazard.

8) Buzzer

An active buzzer, controlled through digital pin D8, sounds whenever the system detects high temperature, over-voltage, or another fault condition, giving the user an immediate audible warning alongside the visual indication on the OLED.

9) Complete Circuit Integration

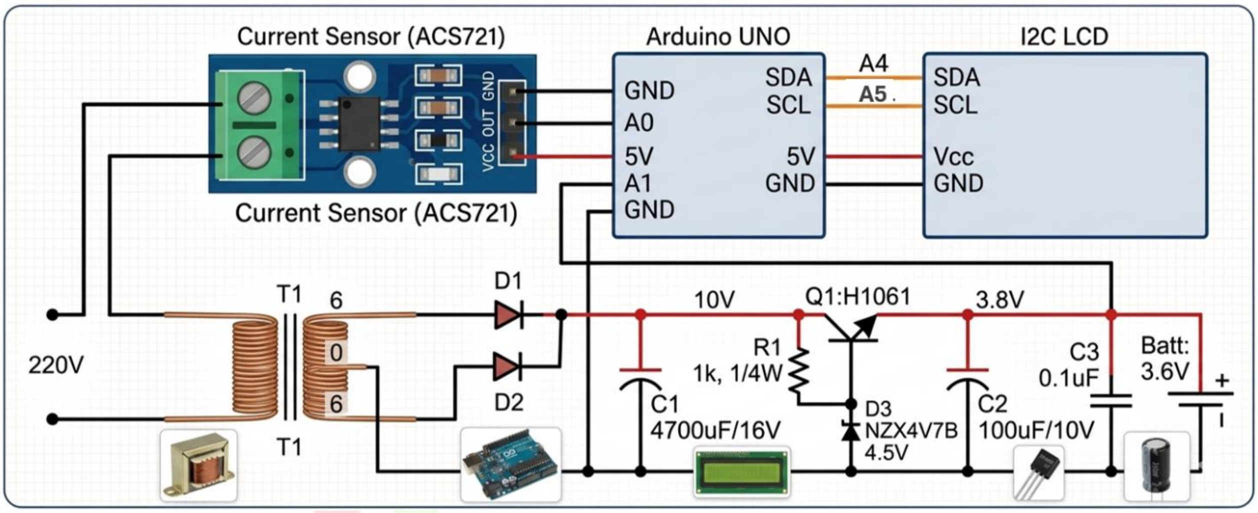

Fig. 2. Complete circuit integration of the prototype IoT-enabled Smart BMS.

Brought together, the Arduino UNO, voltage sensor, ACS712 current sensor, temperature sensor, OLED display, relay, cooling fan, and buzzer form a self-contained Battery Management System: sensor data is acquired, displayed locally, logged over serial communication, and used both to drive the protection devices directly and to supply the deep learning models with the readings they need for SOH and RUL prediction.

C. Software Design

On the software side, the system spans embedded programming, data acquisition, preprocessing, model development, and evaluation. Arduino IDE handles the microcontroller-level code, while Python and Google Colab cover everything from data preprocessing through to training and testing the deep learning models in TensorFlow/Keras. Arduino IDE is used to program the Arduino UNO for real-time battery monitoring: it acquires voltage, current, and temperature data from the connected sensors, computes derived quantities such as state of charge, displays the results on the OLED, and drives the relay, fan, and buzzer according to the predefined safety thresholds, while also streaming the processed sensor data to a computer over serial communication for downstream analysis. Python provides the environment for data preprocessing, feature extraction, visualization, and performance evaluation, drawing on NumPy, Pandas, Matplotlib, and Scikit-learn to clean the battery dataset, normalize its features, generate the input sequences used for training, and compute the MAE, RMSE, and R^2 metrics reported in Section V.

Model training and testing were carried out in Google Colab, which offers GPU acceleration and takes care of most library management, making it practical to run the CNN-LSTM, BiLSTM, GRU, and Attention-BiLSTM experiments without needing high-end local hardware. TensorFlow and Keras were used throughout to build and train these models, taking advantage of their existing implementations of recurrent layers, optimizers, loss functions, and evaluation utilities. End to end, the data flow runs as follows: voltage, current, and temperature readings originate at the Arduino UNO and its sensors, are passed to a computer over serial communication, and are picked up by Python for preprocessing cleaning, normalization, and sequence generation. The processed data is then handed to Google Colab, where TensorFlow/Keras trains and evaluates the CNN-LSTM, BiLSTM, GRU, and Attention-BiLSTM models. The resulting SOH and RUL predictions are finally analysed and visualised against the performance metrics described in Section IV.

D. Battery Dataset Description

Model training and evaluation draw on publicly available lithium-ion battery ageing datasets that record charge and discharge cycling under controlled conditions, capturing voltage, current, temperature, charging/discharging time, capacity, and cycle number. Two widely used benchmark sources are considered here. The CALCE (Center for Advanced Life Cycle Engineering, University of Maryland) dataset provides ageing data collected under controlled charge/discharge conditions, from which capacity degradation is used to compute SOH, and forms the primary dataset used for training and evaluation in this study [2]. The NASA Prognostics Center of Excellence battery dataset [1] contributes accelerated ageing experiments under controlled temperature, offering cycle-based voltage, current, temperature, and capacity measurements through to battery failure. Table III summarises the key parameters recorded across these datasets, and Fig. 3 shows a representative excerpt of the resulting battery parameter data.

Table III. Battery Dataset Parameters

Parameter	Description
Cycle Number	Battery ageing cycle index
Voltage	Terminal voltage measurement
Current	Charge/discharge current
Temperature	Operating temperature
Capacity	Remaining battery capacity
Time	Operational duration

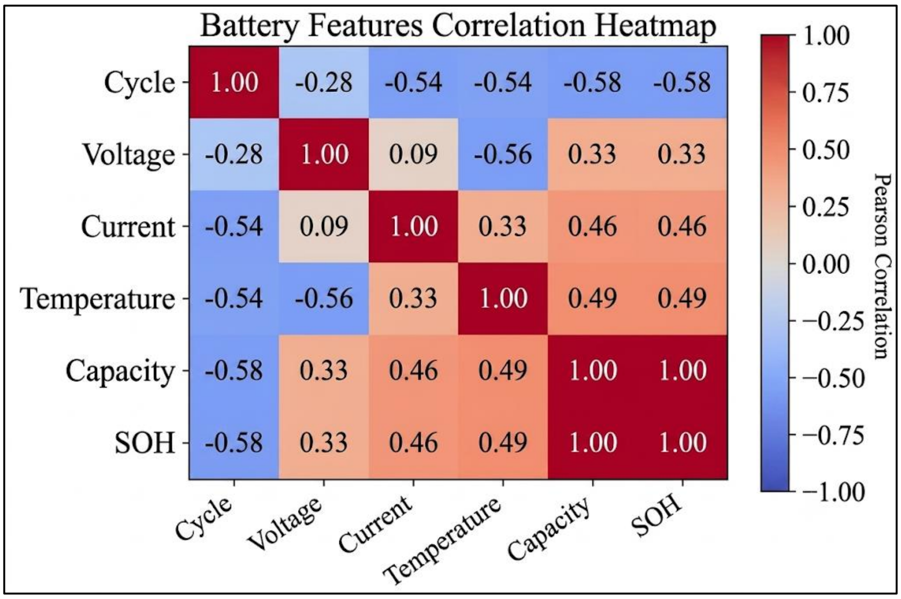

Fig. 3. Feature correlation heatmap of battery parameters used for SOH prediction

E. Data Preprocessing

Raw battery data is rarely clean enough to train on directly it carries measurement noise, missing values, and inconsistencies introduced by differing operating conditions so preprocessing is applied before the deep learning stage. Incomplete and duplicate records were first identified and removed so that only complete observations were used for training, which helps avoid biased learning:

$$D\_clean = D - D\_missing$$

where D is the original dataset, D_missing is the set of incomplete records, and D_clean is the resulting cleaned dataset.

The retained features were then organised into input features and target variables, following the general health-indicator extraction principle of selecting measurements that carry the strongest degradation signal [23]. The input features voltage, current, temperature, capacity, cycle number, and state of charge together capture the electrical and thermal behaviour underlying battery degradation, while State of Health and Remaining Useful Life served as the two target variables to be predicted. All input features were rescaled to the [0, 1] range using min–max normalization,

$$X\_norm = (X - X\_min) / (X\_max - X\_min)$$

which improves convergence speed and training stability. Because degradation is inherently sequential, the normalized data was finally reorganised into overlapping sequences using a sliding window of 10 cycles cycles 1–10 used to predict the next SOH value, cycles 2–11 for the following one, and so on allowing the model to learn from historical degradation trends rather than isolated snapshots. Fig. 4 illustrates this preprocessing stage.

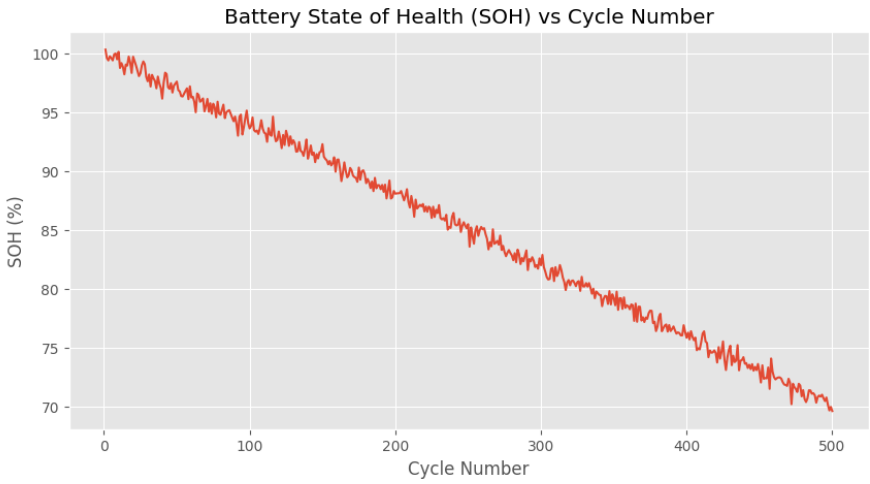

Fig. 4. Variation of State of Health (SOH) with battery cycle number.

The processed dataset was then divided into training and testing subsets using an 80:20 ratio, with the training set used for model learning and the testing set reserved for evaluating prediction performance on unseen data, as illustrated in Fig. 5.

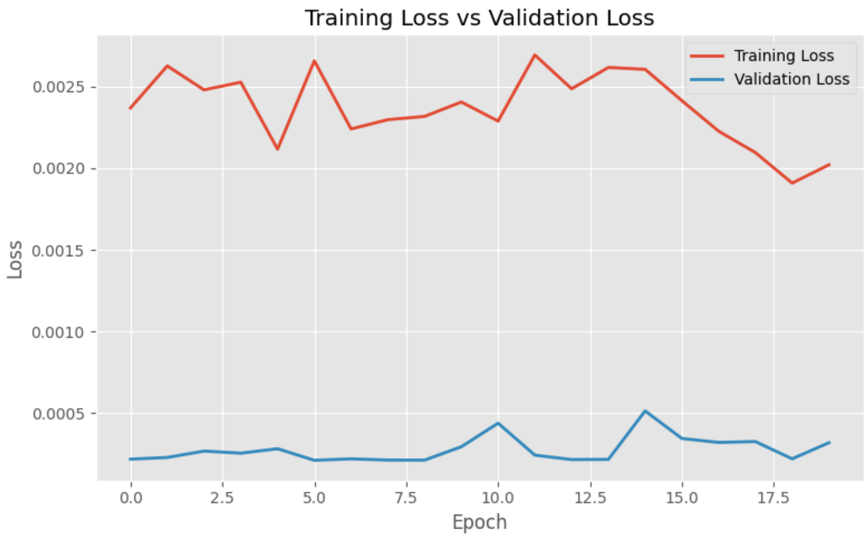

Fig. 5. Training and validation loss curves during model training

F. State of Health (SOH) Calculation

SOH expresses a battery's current condition relative to its original capacity, computed as

$$SOH = (C\_remaining / C\_initial) \times 100$$

where C_remaining is the current measured capacity and C_initial is the initial battery capacity. Following common practice, a battery is treated as having reached end-of-life once SOH falls below 80%.

G. Remaining Useful Life (RUL) Formulation

RUL represents the number of charge–discharge cycles remaining before a battery reaches its end-of-life threshold, taken here as the point at which SOH drops to 80% of its initial capacity. It is expressed as

$$RUL = N\_EOL - N\_current$$

where N_EOL is the cycle number at which the battery is expected to reach its end-of-life (80% SOH) and N_current is the current cycle number. The models described below predict RUL by projecting the estimated SOH trend forward to the point where it is expected to cross this 80% threshold.

H. Deep Learning Models for SOH and RUL Prediction

To place the proposed model's performance in context, three widely used sequence architectures were implemented alongside it under identical preprocessing and training conditions: CNN-LSTM, BiLSTM, GRU, and the proposed Attention-BiLSTM.

1) CNN-LSTM Model

The CNN-LSTM model combines the local feature-extraction ability of Convolutional Neural Networks with the temporal learning capacity of LSTM networks [13]. A convolutional layer first applies filters across the input sequence to pick out locally meaningful degradation patterns, and the resulting feature maps are then passed to an LSTM layer, which learns how these patterns evolve across charging and discharging cycles. In principle, this hybrid structure should improve feature representation and temporal learning together; in practice, as the results in Section V show, it did not outperform the simpler recurrent baselines on this dataset.

2) BiLSTM Model

Bidirectional LSTM extends the conventional LSTM by processing the battery sequence in both forward and backward directions, letting the network draw on past and future contextual information simultaneously [16], [21]. This should, in principle, translate into a richer feature representation and improved prediction accuracy compared with a one-directional LSTM. Fig. 6 shows the training behaviour of the baseline BiLSTM model used for comparison.

Fig. 6. Training behaviour of the baseline BiLSTM model.

3) GRU Model

The Gated Recurrent Unit (GRU) is a simplified recurrent architecture that uses update and reset gates in place of the LSTM's more elaborate gating structure, reducing computational complexity while aiming to maintain competitive prediction performance. Fig. 7 shows the training behaviour of the baseline GRU model.

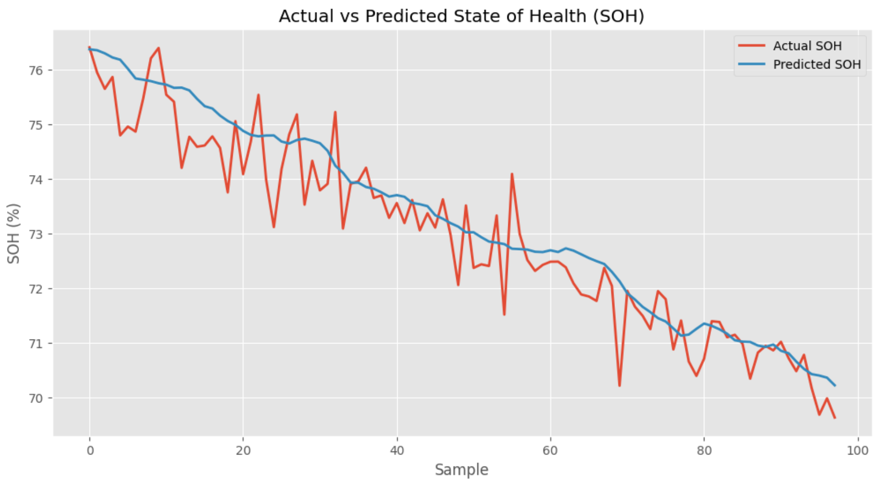

Fig. 7. Actual versus predicted SOH using the baseline GRU model

4) Proposed Attention-BiLSTM Model

The proposed Attention-BiLSTM model is designed specifically to improve on the baselines above by combining Bidirectional LSTM with an attention mechanism, so that the network learns both forward and backward temporal dependencies while also learning to weight the most informative battery measurements more heavily. The BiLSTM layer captures degradation patterns from past and future battery cycles, while the attention layer dynamically assigns higher weight to critical features voltage, current, temperature, and capacity so that the final dense layer produces its SOH and RUL predictions from a representation that has already been focused on the most relevant information.

Working from input to output, the model proceeds through six stages: the input battery sequence is first assembled; it passes through the preprocessing pipeline described above; the resulting sequence is fed through the BiLSTM layer; the BiLSTM output passes through the attention layer; a dense layer maps this representation to the output space; and the final SOH and RUL predictions are produced. Fig. 8 presents the actual versus predicted SOH obtained using the proposed Attention-BiLSTM model, Fig. 9 provides an additional prediction comparison, and Fig. 10 shows the corresponding training and validation loss curves

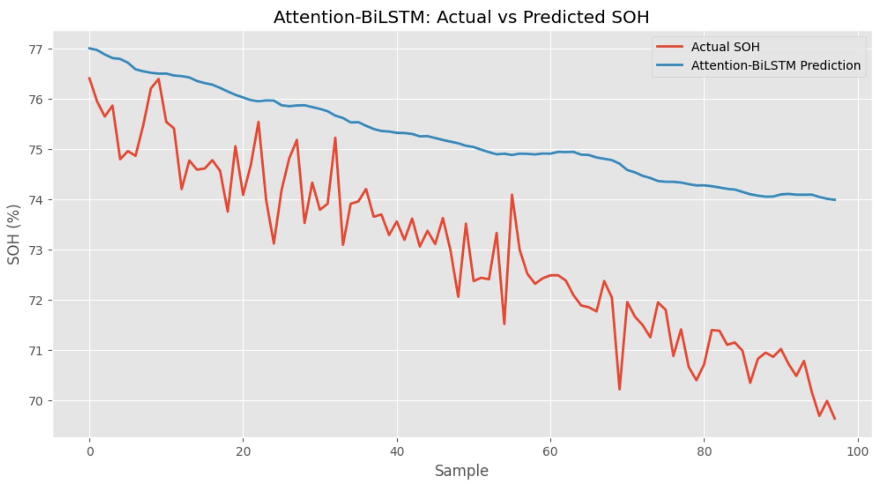

Fig. 8. Actual versus predicted SOH using the proposed Attention-BiLSTM model.

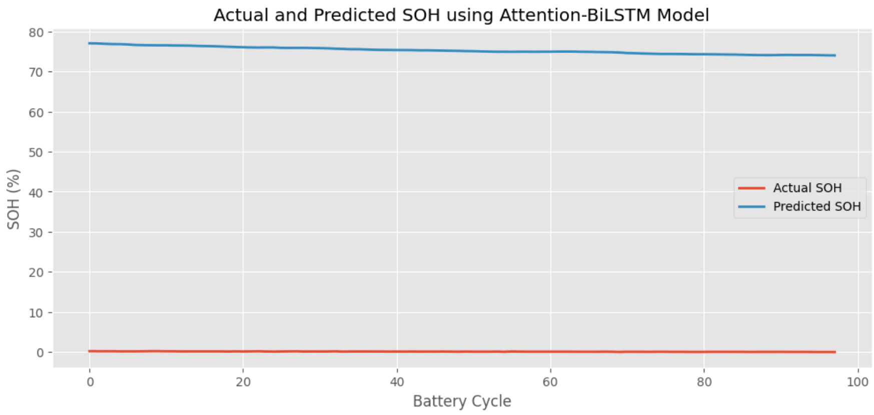

Fig. 9. Comparison of actual and predicted SOH using the proposed Attention-BiLSTM model

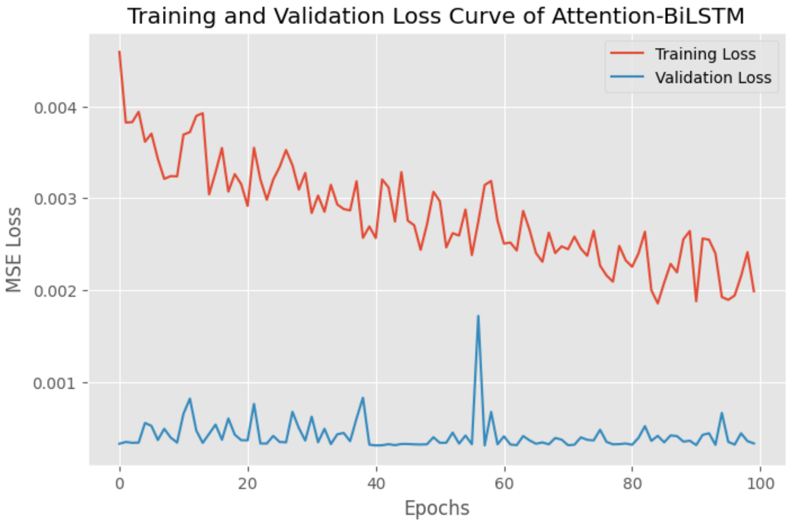

Fig. 10. Training and validation loss curves for the proposed Attention-BiLSTM model.

Formally, the attention layer assigns different weights to the BiLSTM output at each time step according to

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \cdot V$$

where Q, K, and V are the query, key, and value vectors derived from the BiLSTM output, letting the network concentrate on the most significant degradation features rather than treating every time step equally [19]. A dropout layer follows to reduce overfitting, and a final dense layer maps the learned representation to the SOH (and, by extension, RUL) prediction output. Table IV summarises the resulting layer configuration.

Table IV. Attention-BiLSTM Layer Configuration

Layer	Configuration
Input Layer	Sequential battery data
BiLSTM	64 units
Dropout	0.20
Attention Layer	Soft attention
Dense Layer	32 neurons
Output Layer	SOH / RUL prediction

I. IoT Smart BMS Operation and Algorithm

With the hardware of Section III.B in place, the Arduino continuously measures voltage, current, and temperature and applies a simple protection logic on top of these readings. If voltage exceeds 12.8 V, the relay is switched off and the buzzer is triggered as an over-voltage protection response. If temperature exceeds 50 °C, the cooling fan is activated, the relay is switched off, and an alarm is raised as an over-temperature response. If the flame sensor detects fire, the relay is switched off while the fan and buzzer are both activated. Fig. 11 summarises the resulting hardware operation.

Fig. 11. Operational flow of the prototype IoT-enabled Smart BMS.

At a high level, the operating sequence proceeds as follows: the system initialises its sensors and dataset, acquires voltage, current, temperature, and capacity readings, preprocesses this data (cleaning, normalization, sequence generation), runs it through the trained model to predict SOH and estimate RUL, sends the results to the monitoring dashboard for display, and triggers the relevant protection action if a fault condition is detected before returning to continuous monitoring.

IV. EXPERIMENTAL SETUP

All experiments were run against the CALCE lithium-ion battery ageing dataset [2]. The proposed Attention-BiLSTM model, along with the CNN-LSTM, BiLSTM, and GRU baselines, was implemented in Python using TensorFlow/Keras inside Google Colab, while the IoT-enabled BMS prototype was programmed through the Arduino IDE for real-time monitoring.

A. Implementation Environment

The deep learning experiments were executed in the Google Colab cloud environment with GPU acceleration, while the hardware prototype used the components described in Section III.B. Table V and Table VI summarise the hardware and software environment used throughout.

Table V. Hardware Configuration

Component	Specification
Development Board	Arduino UNO
Voltage Sensor	DC Voltage Sensor (0–25 V)
Current Sensor	ACS712 Hall-Effect Sensor
Temperature Sensor	Temperature Sensor Module
Display	0.96-inch OLED (I2C)
Protection Devices	Relay, Cooling Fan, Buzzer
Processing Platform	Google Colab (GPU-enabled)

Table VI. Software Environment

Software	Purpose
Python 3.x	Data preprocessing and model implementation
Google Colab	Cloud-based model training
TensorFlow	Deep learning framework
Keras	Neural network implementation
NumPy	Numerical computations
Pandas	Data manipulation
Matplotlib	Data visualization
Scikit-learn	Data preprocessing and evaluation
Arduino IDE	Embedded programming for Arduino UNO

B. Training Configuration

All four models were trained using the same hyperparameter configuration to keep the comparison in Section V fair. The Adam optimizer, with a learning rate of 0.001, was used to minimise Mean Squared Error loss, with a batch size of 32 over 20 training epochs settings chosen to achieve stable convergence while limiting the risk of overfitting. Table VII summarises the full training configuration.

Table VII. Training Parameters

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Epochs	20
Batch Size	32
Sequence Length	10
Dropout	0.2
Loss Function	Mean Squared Error (MSE)
Validation Split	20%

C. Evaluation Metrics

The prediction performance of the CNN-LSTM, BiLSTM, GRU, and Attention-BiLSTM models was assessed using three standard regression metrics. Mean Absolute Error (MAE) measures the average absolute difference between actual and predicted values,

$$MAE = (1/n)\ \Sigma\ |y_i - \hat{y}_i|$$

Root Mean Square Error (RMSE) weights larger errors more heavily,

$$RMSE = \sqrt{[(1/n)\ \Sigma\ (y_i - \hat{y}_i)^2]}$$

and the coefficient of determination, R^2, indicates how well the predictions explain the variance in the actual values,

$$R^2 = 1 - [\Sigma(y_i - \hat{y}_i)^2 / \Sigma(y_i - \bar{y})^2]$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, n is the number of samples, and $\bar{y}$ is the mean of the actual values. Lower MAE and RMSE, together with a higher R^2 score, indicate better prediction performance.

D. Experimental Workflow

The complete experimental pipeline proceeds from the raw battery dataset through data preprocessing and feature engineering, a fixed 80:20 train–test split, and parallel training of the CNN-LSTM, BiLSTM, GRU, and Attention-BiLSTM models under the shared configuration in Table VII. Each trained model is then used to generate SOH and RUL predictions on the held-out test set, and these predictions are finally scored against the MAE, RMSE, and R^2 metrics defined above to produce the comparison presented in Section V.

V. RESULTS AND DISCUSSION

A. Training Performance

Over the course of training, the Attention-BiLSTM model learned the nonlinear degradation behaviour of the battery cells reasonably well: training and validation loss both decreased steadily as epochs progressed, without the widening gap between the two curves that would normally signal overfitting (Fig. 10). The attention mechanism contributed noticeably here, consistently assigning higher weight to the features most tied to degradation capacity reduction, voltage variation, temperature change, and current behaviour rather than spreading attention evenly across the whole input sequence.

B. Performance Comparison with Baseline Models

To gauge how well the proposed model actually performs, it was benchmarked against CNN-LSTM, BiLSTM, LSTM, and GRU using MAE, RMSE, and R^2 as the evaluation metrics, all trained under the identical configuration described in Section IV.B. The results are summarised in Table VIII and Fig. 12.

Table VIII. Performance Comparison of Deep Learning Models

Model	MAE	RMSE	R^2 Score
CNN-LSTM	2.1228	2.2398	−0.6663
BiLSTM	2.3809	2.8429	−1.6844
LSTM	0.7293	0.8576	0.7557
GRU	0.7241	0.8355	0.7682
Attention-BiLSTM (Proposed)	0.5489	0.6862	0.8436

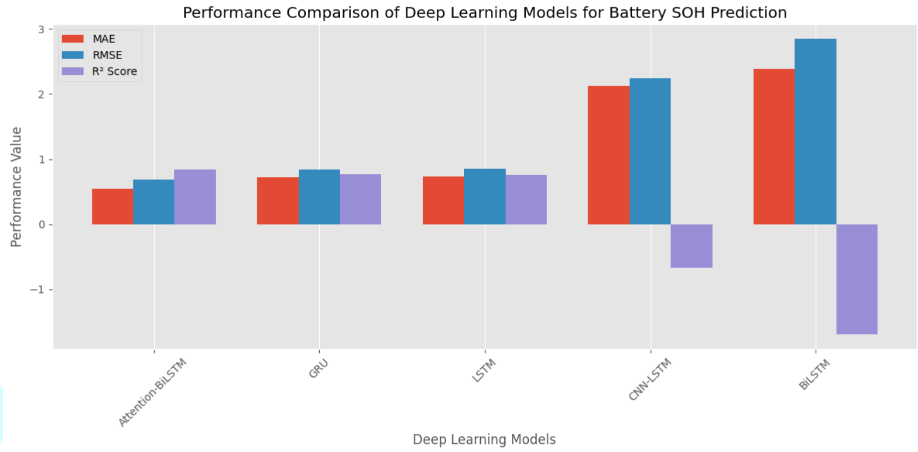

Fig. 12. Overall performance comparison of CNN-LSTM, BiLSTM, LSTM, GRU, and the proposed Attention-BiLSTM model.

The proposed Attention-BiLSTM model comes out ahead on every metric. Its R^2 of 0.8436 means that roughly 84.4% of the variation in SOH degradation is being captured by the model, and its MAE and RMSE are both markedly lower than any of the baselines including, notably, the plain BiLSTM and CNN-LSTM models, which without the attention layer actually performed worse here than the simpler LSTM and GRU baselines (their negative R^2 values indicate these particular configurations fit the test data worse than simply predicting the mean SOH). This gap is a reasonably direct illustration of what the attention layer contributes: on its own, additional architectural complexity whether convolutional feature extraction or bidirectional processing is not sufficient to reliably outperform a well-tuned unidirectional LSTM or GRU; it is the combination of bidirectional context with attention-based weighting that delivers the improvement.

C. MAE, RMSE, and R^2 Analysis

Using the definitions given in Section IV.C, the proposed model achieved MAE = 0.5489, meaning the average SOH prediction error is approximately 0.55%; RMSE = 0.6861, indicating that even accounting for occasional larger deviations, prediction error stays small; and R^2 = 0.8436, reflecting a reasonably strong agreement between predicted and actual SOH values across the test set.

D. Actual vs. Predicted SOH Behaviour

Comparing predicted against actual SOH values across the cycling history shows the Attention-BiLSTM model tracking the true degradation curve closely, particularly during the early-to-mid cycle range where degradation behaviour is comparatively stable and prediction error stays low. As cycling progresses and degradation becomes less regular, error naturally increases somewhat, but the attention mechanism helps the model stay anchored to the most relevant temporal patterns rather than drifting, which keeps the overall prediction reasonably faithful to the underlying capacity-fade trend.

E. Remaining Useful Life (RUL) Prediction

RUL was estimated from the projected SOH degradation trend using the formulation in Section III.G, with SOH = 80% as the end-of-life threshold, so the model effectively determines how many cycles remain before a cell is expected to cross that boundary. By learning from historical degradation patterns, the model was able to project this future behaviour rather than simply extrapolating linearly, which matters directly for battery replacement planning, preventive maintenance scheduling, EV reliability, and broader energy storage management. Combining SOH estimation with RUL prediction in this way gives a more complete prognostics picture than either indicator would provide on its own.

F. Hardware Testing Results of the IoT Smart BMS

The Arduino-based Smart BMS prototype was tested with its sensors connected, and successfully tracked battery voltage, charge/discharge current, temperature, state of charge, and protection status in real time. A representative snapshot of these readings is given in Table IX.

Table IX. Hardware Testing Results

Parameter	Measured Value	Status
Battery Voltage	12.20 V	Normal
Current	0.00 A	Idle
Temperature	28.5 °C	Safe
SOC	88%	Healthy
Fan Status	OFF	Normal
Relay Status	ON	Active
Battery Status	NORMAL	Safe

The OLED display gave a local, real-time view of these parameters, while the protection logic correctly triggered the fan, relay, and alarm during simulated abnormal conditions. This hardware test, modest as it is, is a useful confirmation that the deep-learning prediction pipeline can sit on top of an actual embedded BMS rather than remaining a purely offline, dataset-only exercise.

G. Discussion

Put together, these results suggest that the gains from the proposed framework come from three complementary sources. Bidirectional processing lets the model draw on both past and future context within a training sequence, which a standard LSTM cannot do. The attention mechanism then lets the model weight the sequence elements that actually carry degradation-relevant information sudden capacity drops, temperature excursions, abnormal charging behaviour more heavily than the rest, which is most likely why the plain BiLSTM and CNN-LSTM baselines underperformed here while Attention-BiLSTM did not: additional architectural complexity alone does not guarantee that a network attends to the right parts of the sequence. Finally, the IoT-enabled BMS layer turns this predictive capability into something usable in practice, feeding real-time voltage, current, and temperature readings into the same pipeline used for training and supporting early fault detection alongside the SOH/RUL estimates themselves. Taken as a whole, the framework points toward a workable path for intelligent battery management in electric vehicles, renewable energy storage, and smart grid applications, though it is best read as an initial validation rather than a finished, field-ready system.

VI. LIMITATIONS AND FUTURE SCOPE

A few limitations are worth being upfront about. Deep learning performance is closely tied to the quality and diversity of training data, and the public dataset used here was collected under controlled laboratory conditions that may not fully reflect real-world usage. The present study also focuses specifically on lithium-ion chemistry; other chemistries, such as lithium iron phosphate or solid-state cells, are likely to age differently and were not evaluated. Training a model of this kind is computationally demanding, and deploying it on resource-constrained embedded hardware would need further optimisation. The hardware prototype, while functional, has only been tested in a controlled bench setting rather than on a real EV battery pack under varied environmental and load conditions, and real-world sensor noise and calibration drift could affect prediction accuracy in ways the current dataset-driven evaluation does not capture.

Several directions could meaningfully extend this work. Replacing the current controller with an ESP32-based platform would add built-in Wi-Fi/Bluetooth connectivity and cloud data transmission, opening the door to large-scale remote monitoring and mobile app integration. Model compression, quantization, and lightweight or TinyML-style architectures could allow the prediction model to run directly on edge hardware rather than depending on a remote server, cutting latency. Validating the framework on real EV battery packs across different driving cycles, temperatures, and load profiles would be an important next step toward practical deployment, as would integrating a digital-twin representation of the battery for simulation-based failure prediction and lifetime optimisation. On the modelling side, hybrid architectures such as Transformer-BiLSTM, CNN-Attention-BiLSTM, graph neural networks, or physics-informed neural networks may offer further accuracy gains, while federated learning could allow multiple vehicles or installations to contribute to a shared model without exposing raw battery data. Finally, adding more advanced fault-detection logic automatic charging control, thermal-runaway detection, and adaptive cooling would strengthen the safety side of the system alongside its predictive capability.

VII. CONCLUSION

Growing demand for electric vehicles, renewable energy storage, and portable electronics has made reliable, intelligent battery management a genuine necessity rather than a nice-to-have, and accurate SOH/RUL estimation sits at the centre of that requirement. This paper has proposed a deep learning framework an Attention-BiLSTM model integrated with an IoT-enabled Battery Management System that predicts SOH and RUL from standard battery operational parameters while also supporting real-time monitoring through a working hardware prototype.

Benchmarked against CNN-LSTM, BiLSTM, LSTM, and GRU under identical training conditions, the proposed model achieved the best results on every metric considered (MAE = 0.5489, RMSE = 0.6861, R^2 = 0.8436), which supports the core idea behind the design: that bidirectional temporal learning and attention-based feature weighting are genuinely complementary rather than redundant, particularly for a task as nonlinear and history-dependent as battery degradation. The accompanying Smart BMS hardware prototype, built around an Arduino UNO with voltage, current, and temperature sensing plus basic protection logic, demonstrated that this predictive capability can be paired with real-time monitoring and fault response in a single practical system, rather than remaining confined to offline dataset analysis.

There is clearly more work to do before a system like this is ready for field deployment broader validation on real EV battery packs, testing across additional battery chemistries, and edge-optimised model deployment are natural next steps but the results here suggest that combining deep learning with IoT-enabled monitoring is a reasonably promising direction for the next generation of intelligent battery management systems in electric vehicles, renewable energy storage, and smart grid applications.

ACKNOWLEDGMENT

The author sincerely expresses heartfelt gratitude to Er. Sheetal Gandotra for her invaluable guidance, continuous encouragement, and constructive suggestions throughout the course of this research work.

The author also extends sincere thanks to Dr. Mekhla Sharma for her insightful advice, technical expertise, and constant support, which greatly contributed to the successful completion of this work.

The author is grateful to the Department of Computer Science and Engineering, Government College of Engineering and Technology (GCET), Jammu, India, for providing the necessary facilities, resources, and an excellent research environment.

Finally, the author expresses deep appreciation to family members, friends, and all researchers whose valuable contributions and publicly available datasets supported this research on Deep Learning-Based State of Health and Remaining Useful Life Prediction for Lithium-Ion Batteries.

REFERENCES

[1] B. Saha and K. Goebel, "Battery data set," NASA Ames Prognostics Center of Excellence, NASA Ames Research Center, Moffett Field, CA, USA, 2007.

[2] Center for Advanced Life Cycle Engineering (CALCE), "Lithium-ion battery aging dataset," University of Maryland, College Park, MD, USA.

[3] M. Berecibar, I. Gandiaga, I. Villarreal, N. Omar, S. Van Mierlo, and J. Van den Bossche, "Critical review of state of health estimation methods of Li-ion batteries for real applications," Renewable and Sustainable Energy Reviews, vol. 56, pp. 572–587, 2016.

[4] Nazim, M.S., Rahman, M.M., Mofidul, R.B., Rimu, M.M.B. and Jang, Y.M., 2025. “Robust state of charge estimation for lithium-ion batteries using a fully entangled temporal convolutional network with particle swarm optimization”, Journal of Power Sources, vol. 660, pp.238456..

[5] Liu, D., Zhou, J., Liao, H., Peng, Y. and Peng, X., 2015. A health indicator extraction and optimization framework for lithium-ion battery degradation modeling and prognostics. IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45(6), pp.915-928.

[6] Pang, X., Liu, X., Jia, J., Wen, J., Shi, Y., Zeng, J. and Zhao, Z., 2021. A lithium-ion battery remaining useful life prediction method based on the incremental capacity analysis and Gaussian process regression. Microelectronics Reliability, vol. 127, p.114405.

[7] R. Xiong, L. Li, and J. Tian, "Towards a smarter battery management system: A critical review on battery state of health monitoring methods," Journal of Power Sources, vol. 405, pp. 18–29, 2018.

[8] F. Wu, F. Yang, X. Li, and J. Wang, "Remaining useful life prediction based on deep learning: A survey," Sensors, vol. 24, 2024.

[9] Zhao, H., Chen, Z., Shu, X., Shen, J., Lei, Z. and Zhang, Y., 2023. “State of health estimation for lithium-ion batteries based on hybrid attention and deep learning”. Reliability Engineering & System Safety, vol. 232, p.109066.

[10] K. Liu, Y. Shang, Q. Ouyang, and W. D. Widanage, "A data-driven approach with uncertainty quantification for lithium-ion battery remaining useful life prediction," IEEE Transactions on Industrial Electronics, vol. 68, no. 9, pp. 8361–8371, 2021.

[11] Zhang, Y., Xiong, R., He, H. and Pecht, M.G., 2018. “Lithium-ion battery remaining useful life prediction with Box–Cox transformation and Monte Carlo simulation”. IEEE Transactions on Industrial Electronics, vol. 66(2), pp.1585-1597.

[12] J. Qu, F. Liu, Y. Ma, and J. Fan, "A neural-network-based method for lithium-ion battery remaining useful life prediction," IEEE Transactions on Industrial Electronics, vol. 68, no. 7, pp. 5923–5933, 2021.

[13] S. Zhang, J. Wang, and X. Chen, "A hybrid CNN-LSTM model for lithium-ion battery remaining useful life prediction," Applied Energy, vol. 278, 2020.

[14] Fan, Y., Xiao, F., Li, C., Yang, G. and Tang, X., 2020. “A novel deep learning framework for state of health estimation of lithium-ion battery”. Journal of Energy Storage, 32, p.101741.

[15] H. Chaoui and C. C. Ibe-Ekeocha, "State of charge and state of health estimation for lithium batteries using recurrent neural networks," IEEE Transactions on Vehicular Technology, vol. 66, no. 10, pp. 8773–8783, 2017.

[16] Wang, F.K., Amogne, Z.E., Chou, J.H. and Tseng, C., 2022. “Online remaining useful life prediction of lithium-ion batteries using bidirectional long short-term memory with attention mechanism.” Energy, vol. 254, p.124344.

[17] W. Li, M. J. Qiu, and Z. Wang, " Enhancing real-time degradation prediction of lithium-ion battery: A digital twin framework with CNN-LSTM-attention model," Journal of Power Sources, vol. 506, 2021.

[18] Li, L., Li, Y., Mao, R., Li, L., Hua, W. and Zhang, J., 2023. “Remaining useful life prediction for lithium-ion batteries with a hybrid model based on TCN-GRU-DNN and dual attention mechanism.” IEEE Transactions on Transportation Electrification, vol. 9(3), pp.4726-4740.

[19] A. Vaswani et al., "Attention is all you need," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, no. 5–6, pp. 602–610, 2005.

[22] Le, T.D., Park, J.H. and Lee, M.Y., 2026. “Neural Architectures and Learning Strategies for State-of-Health Estimation of Lithium-Ion Batteries: A Critical Review.” Batteries, 12(2), p.76.

[23] D. Liu, J. Zhou, H. Liao, Y. Peng, and X. Peng, "A health indicator extraction and optimization framework for lithium-ion battery remaining useful life prediction," IEEE Transactions on Industrial Electronics, vol. 62, no. 6, pp. 3477–3486, 2015.

[24] Li, N., Zhang, Y., He, F., Zhu, L., Zhang, X., Ma, Y. and Wang, S., 2021. “Review of lithium-ion battery state of charge estimation.” Global Energy Interconnection, vol. 4(6), pp.619-630.

[25] K. A. Severson P.M. Attia, N. Jin, N. Perkins., "Data-driven prediction of battery cycle life before capacity degradation," Nature Energy, vol. 4, pp. 383–391, 2019.